

UNITED STATES PATENT APPLICATION
FOR

SYNTHESIZING COHERENT CORRELATION SUMS AT ONE OR MULTIPLE
CARRIER FREQUENCIES USING CORRELATION SUMS
CALCULATED AT A COARSE SET OF FREQUENCIES

INVENTORS:

ANANT SAHAI
JOHN TSITSIKLIS

PREPARED BY:

HICKMAN PALERMO TRUONG & BECKER LLP
1600 WILLOW STREET
SAN JOSE, CALIFORNIA 95125
(408) 414-1080

EXPRESS MAIL CERTIFICATE OF MAILING

"Express Mail" mailing label number

EL734779482US

Date of Deposit

JUNE 22, 2001

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

(Typed or printed name of person mailing paper or fee)

Conia M. Tan

(Signature of person mailing paper or fee)

**SYNTHESIZING COHERENT CORRELATION SUMS AT ONE OR
MULTIPLE CARRIER FREQUENCIES USING CORRELATION SUMS
CALCULATED AT A COARSE SET OF FREQUENCIES**

FIELD OF THE INVENTION

5 The present invention relates to signal processing and, more particularly, to techniques for synthesizing coherent correlation sums at a fine frequency resolution using coherent correlation sums that are calculated at a coarse frequency resolution.

BACKGROUND OF THE INVENTION

In general, when a carrier-modulated signal is received by a receiver, the carrier frequency of the signal is shifted by an unknown amount, due to clock drift at the receiver and/or a Doppler shift due to motion. Further, the known signal may be delayed by an unknown amount of time in the course of traveling from the source to the receiver. The signal that is received at the receiver is generally referred to as the received signal, and it typically includes noise. For many applications, one needs to determine the shift of the carrier frequency to a high degree of accuracy in order to be able to estimate some of the signal parameters such as the signal delay or the data that are being communicated through the signal.

In one approach for estimating the carrier frequency, known as coherent processing, multiple candidate carrier frequencies are examined. For each candidate frequency, In Phase (“I”) and Quadrature (“Q”) correlation sums are calculated, followed by an evaluation of 20 the sum of squares $I^2 + Q^2$. The candidate frequency that results in the largest sum of squares is selected as the estimated carrier frequency. If the delay is an unknown quantity, then a two-dimensional search is performed to simultaneously estimate the delay and carrier

frequency.

A significant disadvantage to the above approach is that the computational expense in performing the calculations for the various candidate frequencies can be prohibitively high. This is because the number of candidate frequencies that have to be examined must increase 5 in proportion to the duration of the signal and therefore in proportion to the number N of available samples of the received signal. Consequently, the overall computational effort is of the order of at least N^2 .

In another approach, the signal is broken into multiple short blocks, each block is processed coherently as above, and then the values of $I^2 + Q^2$ calculated from the different blocks 10 are added. This approach, known as non-coherent processing, can work with a smaller number of candidate frequencies, compared to coherent processing. On the other hand, it is much more sensitive to noise. Therefore, in order to maintain a constant level of performance, more signal samples are required and the overall computational effort remains high.

Based on the foregoing, there is a clear need for a Technique for synthesizing coherent 15 correlation sums at a fine frequency resolution using coherent correlation sums that are calculated at a coarse frequency resolution that requires less computational expense.

PROBLEM STATEMENT

The general problem of estimating an unknown shift in the carrier frequency of a received signal is as follows. Given a sequence of N real numbers, denoted by x_1, \dots, x_N , the following 20 expression is to be evaluated (exactly or approximately):

$$C(f) = \sum_{k=1}^N x_k e^{2\pi j f k \Delta}, \quad (1)$$

for a plurality of frequencies f in a given range

$$F_1 \leq f \leq F_2.$$

Herein, j is used to denote the square root of -1 , that is, $j = \sqrt{-1}$. Also, Δ is a known

constant that represents the spacing in time between consecutive numbers x_k . Note that

5 $C(f)$ is complex-valued. The real and imaginary parts of $C(f)$ are known as In Phase (I) and Quadrature (Q) correlation sums, respectively.

In a variant of the problem, it is desired to maximize $|C(f)|^2$, commonly referred to as the “ambiguity function,” over the given frequency range. This maximization problem is typically solved by evaluating $|C(f)|^2$ over a densely-spaced grid of candidate values of f ,
10 and selecting that value of f for which $|C(f)|^2$ is largest. This procedure finds the maximum over the chosen grid. However, when the grid spacing is small and the function $C(f)$ is smooth, the results of this search provide a very close approximation to the search over the entire range $[F_1, F_2]$ of frequencies. In another variant, one may seek to identify values of f for which $|C(f)|^2$ exceeds a certain threshold.

15 In a further variant of the problem, the numbers x_k depend, in a known manner, on an additional unknown parameter σ , and are denoted by $x_k(\sigma)$ to make this dependence explicit. In this variant, the problem consists of calculating the correlation sum

$$C(f, \sigma) = \sum_{k=1}^N x_k(\sigma) e^{2\pi j f k \Delta}, \quad (2)$$

for various values of the pair (f, σ) or of maximizing $C(f, \sigma)$ over a given range of (f, σ)

20 pairs.

In another variant, the parameter σ can take integer values within a given range $\mathcal{M} = \{0, 1, 2, \dots, M - 1\}$ and $x_k(\sigma)$ is of the form

$$x_k = y_k s_{k-\sigma},$$

for some known sequences y_k and s_l , resulting in

5 $C(f, \sigma) = \sum_{k=1}^N y_k s_{k-\sigma} e^{2\pi j f k \Delta}. \quad (3)$

In a special case of the problem one may wish to calculate the correlation sum for a single choice or for very few choices of f , but do this in a computationally efficient manner that makes use of correlation sums that have already been computed for frequencies that differ from the frequencies that are of interest. This situation can arise in a number of ways. In some cases, the signal is broken into blocks and correlation sums are automatically computed at the receiver, e.g., by a bank of analog or digital correlators, at a preset set of frequencies. One may want to make computationally efficient use of the available block correlation sums to synthesize the desired correlation sums over the entire duration of the signal. In some cases, the raw data may be unavailable, so that such a synthesis of long correlation sums on the basis of short correlation sums is the only option available. In other cases, an algorithm may have already computed correlation sums for certain values of f , and of particular interest are the correlation sums for a nearby value of f .

As another variant, one may want to compute a correlation sum of a more general form, such as

20 $C(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j f k \Delta + j \theta_k}, \quad (4)$

or

$$C(f, \sigma) = \sum_{k=1}^N y_k s_{k-\sigma} e^{2\pi j f k \Delta + j \theta_k}, \quad (5)$$

where θ_k is a known phase term that changes slowly with time.

ORIGINS OF THE PROBLEM

5 The above stated problem arises during the acquisition of a carrier-modulated signal that has been distorted by an unknown shift of its carrier frequency. In this subsection, a summary description of this particular context is provided.

A receiver receives a signal of the form

$$y(t) = \alpha s(t; \sigma) \cos(2\pi(f_0 + f^*)t + \theta(t) + \phi^*) + w(t), \quad t \in [0, T].$$

10 Here, t stands for time, T is the total duration of the signal, f_0 is a known carrier frequency, f^* is an unknown shift of the carrier frequency, $\theta(t)$ is an additional known phase term, ϕ^* is the typically unknown phase of the modulating signal, and $w(t)$ represents noise. Furthermore, α is a usually unknown constant which is proportional to the square root of the power of the received signal, and $s(t; \sigma)$ stands for the baseband form of the received signal, in the
15 absence of noise. Finally, σ is an unknown parameter that determines the exact form of the signal in the absence of noise. The form of the dependence on σ is assumed to be known. That is, if σ were known, then the value of $s(t; \sigma)$ would be known for every t .

In a digital communications context, $s(t; \sigma)$ can be a transmitted waveform, and σ can represent the data bits that are being communicated through this waveform. In wireless
20 and satellite communications, including Global Positioning Systems (GPS), σ can represent an unknown time delay from the transmitter to the receiver. By letting σ be the ratio of

the delay to the intersample time Δ , and if this ratio is integer, one obtains that $s(t; \sigma)$ is actually of the form $s(t - \sigma\Delta)$.

The variable f^* represents an unknown frequency shift, which is known to belong to certain range $[F_1, F_2]$. This frequency shift can arise in various ways. A representative 5 but not exclusive list includes receiver motion, transmitter motion, or a clock skew at the transmitter or receiver. A frequency shift due to motion is known as a Doppler shift.

The additional but known phase term $\theta(t)$ can arise in a situation where the transmitter or receiver's velocity is changing in a predictable manner. For a specific context, consider a GPS receiver that is operating on data that have been collected over a 1 second interval.

10 During that interval, a satellite that transmits a GPS signal moves considerably, and the satellite motion manifests itself as a Doppler shift of the carrier frequency. In addition, because the satellite does not move on a straight line relative to a stationary receiver but rather follows an elliptical orbit, the velocity of the satellite can change appreciably during the 1 second interval. Such a change in velocity (acceleration) can be modeled as a change 15 in the carrier frequency or equivalently as a slowly changing additional phase term $\theta(t)$.

Because the satellite orbit is known, the additional phase term $\theta(t)$ is also known.

In some contexts, the signal is immediately mixed at the receiver from the carrier frequency to an intermediate frequency, in which case the notation f_0 in the above formula can be taken to represent the intermediate frequency. In some contexts, the signal is immediately 20 mixed directly to baseband, in which case f_0 can be taken equal to zero.

In some cases, the received signal $y(t)$ is analog and is subsequently filtered and sampled. In other cases, the received signal is obtained from an intermediary device in filtered and

sampled form. In either case, the result is a discrete sequence of samples of the form

$$y_k = s_k(\sigma) \cos(2\pi(f_0 + f^*)k\Delta + \theta_k + \phi^*) + w_k.$$

Here, y_k , $s_k(\sigma)$, θ_k , and w_k reflect the value of the functions $y(t)$, $s(t; \sigma)$, $\theta(t)$ and $w(t)$, or of filtered versions of these functions, at time $k\Delta$, where Δ is the length of time that elapses

5 between two consecutive samples. Without loss of generality, it is assumed here that the time at which the first sample is taken is Δ , and that T is an integer multiple of Δ . Let N be the total number of samples, which is given by

$$N = \frac{T}{\Delta}.$$

A coherent receiver operates as follows. It calculates In-Phase and Quadrature correlation

10 sums of the form

$$I(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) \cos(2\pi f k \Delta + \theta_k),$$

and

$$Q(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) \sin(2\pi f k \Delta + \theta_k),$$

for various pairs (f, σ) , and searches for a pair that results in the largest value of the ambi-

15 guity function

$$I^2(f, \sigma) + Q^2(f, \sigma).$$

One now defines $x_k = y_k s_k(\sigma)$, and $C(f, \sigma) = I(f, \sigma) + jQ(f, \sigma)$. One then observes that coherent identification of the carrier frequency for a particular candidate value of σ coincides with the problem of maximizing $|C(f, \sigma)|^2$, with $C(f, \sigma)$ defined in Eq. (4), that is,

$$C(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j f k \Delta + j \theta_k}.$$

20

When σ is the delay in the reception of the signal, that is, if $s(t; \sigma)$ is of the form $s(t - \sigma\Delta)$, then $s_k(\sigma)$ is of the form $s_k(\sigma) = s_{k-\sigma}$. It follows that coherent identification of the carrier frequency for a particular candidate value of σ coincides with the problem of maximizing $|C(f, \sigma)|^2$, with $C(f, \sigma)$ defined in Eq. (5), that is,

$$C(f, \sigma) = \sum_{k=1}^N y_k s_{k-\sigma} e^{2\pi j f k \Delta + j \theta_k},$$

When acquiring a signal, one typically needs to perform a two-dimensional search over a set of candidate pairs (f, σ) . If the ambiguity function $|C(f, \sigma)|^2$ is large for a particular (f, σ) , this is taken as evidence that this particular pair provides a close approximation to the true frequency shift and parameter of the received signal, and the signal is said to be detected or acquired. It is common to combine this approach with a thresholding scheme whereby acquisition is deemed to be successful if the ambiguity function exceeds some threshold.

In some cases, the received signal is analog but is not sampled. The processing of the signal can be carried out in analog form, using for instance analog correlators implemented in hardware. In such cases, all of our previous discussion remains applicable, provided that the sums are replaced throughout by integrals. For ease of exposition, our discussion will focus on the case where the signal is sampled and sums are calculated. However, one skilled in the art would have no difficulty applying this invention to the case where sums are replaced by integrals.

TRADITIONAL METHODS

In this section, the methods that are traditionally used for carrier frequency estimation are discussed, and their shortcomings are indicated. Before doing so, the number of candidate

frequencies that need to be examined are discussed.

THE NUMBER OF CANDIDATE FREQUENCIES

Since there are infinitely many different frequencies in the range $[F_1, F_2]$, this range has to be discretized so that $C(f, \sigma)$ is only computed for a finite number of candidate frequencies 5 f , namely those that belong to a suitably constructed grid of frequencies.

Let us take the spacing between successive grid points to be a number Δ_f . One then computes $C(f, \sigma)$ only for those f of the form $f = F_1 + b\Delta_f$, where b is an integer taking values in the range $b = 0, 1, \dots, B$. The number B is chosen so that $F_1 + B\Delta_f \geq F_2$ and is approximately equal to

$$B \approx \frac{F_2 - F_1}{\Delta_f}.$$

With this choice, there is a guarantee that at least one grid point will be within $\Delta_f/2$ from the true frequency f^* .

When a grid frequency f is used in place of the true frequency f^* , the phase term $e^{2\pi jfk\Delta}$ will be different from $e^{2\pi jf^*k\Delta}$. The difference must be kept small enough in order to maintain 15 coherence. For this, it is necessary and sufficient to choose the grid-spacing Δ_f so that $\Delta_f T$ is smaller than some user-selected threshold $1/n$. By setting

$$\Delta_f T = \frac{1}{n}, \tag{6}$$

it is obtained that the number B of frequency grid points is approximately

$$B \approx \frac{F_2 - F_1}{\Delta_f} = n(F_2 - F_1)T = n(F_2 - F_1)\Delta N.$$

For any given choice of n , it is important to note that the number of frequency grid points increases in proportion to the number N of data samples.

A reasonable choice is to let $n = 4$, which results in a maximum phase error (loss of coherence) smaller than $\pi/4$, although other choices are also appropriate. When using a 5 larger value for n , accuracy is somewhat improved, but computational requirements increase because of the larger number B of frequency grid points.

THE STANDARD METHOD

The most common approach is to evaluate the expression for $C(f, \sigma)$ separately for each one of the B frequency grid points. If one counts one multiplication and one addition of complex numbers as one operation, this approach requires about NB operations. Since B increases in proportion to N , the computational requirements of this approach are of the order of N^2 .

If $C(f, \sigma)$ is to be evaluated for M different values of σ , the above procedure has to be repeated M times, for a total computational effort of MBN .

15 To put this in perspective, consider the context where 1 second of a received GPS signal, sampled at 4MHz, is searched. With a frequency shift uncertainty of 1KHz, and with $n = 4$, one has $B = 4,000$, and $N = 4 \times 10^6$. If one searches for $M = 1000$ possible values of the signal delay, the computational requirements amount to 16×10^{12} , just to acquire the signal from a single satellite, which is prohibitively large.

20 In the further special case where the data x_k are of the form $x_k = y_k s_{k-\sigma}$, some savings are possible. One recognizes that $C(f, \sigma)$ is the convolution of two sequences, and that the convolution only needs to be computed for M different choices of σ . One can divide the

sequences into blocks of length M , calculate convolutions block-by-block and add them at the end. The convolution of two sequences of length M can be calculated by forming the transform of the two sequences, using a Fast Fourier Transform (FFT) algorithm, multiplying the transforms, and then taking an inverse transform. This takes $3cM \log M$ arithmetic operations, where $cM \log M$ is the running time of a single FFT or inverse FFT.

The constant c depends on the implementation of the algorithm, but can be assumed to be less than 10. The overall complexity, summing over N/M blocks, and over B frequencies is $3cNB \log M$. Here and throughout the rest of this document, all logarithms will be taken with respect to base 2.

In all cases, and because B increases linearly with N , the overall computational effort grows as N^2 .

A TRANSFORM-BASED METHOD

There is an alternative to the standard method, which is applicable to the case where the additional time-dependent phase term θ_k is absent, and which is the subject of US patent 15 4,701,934, entitled, “Method of Doppler Searching in A Digital GPS Receiver” by inventor Steven C. Jasper. One notes, that for any fixed σ , the expression for $C(f, \sigma)$

$$C(f, \sigma) = \sum_{k=1}^N x_k(\sigma) e^{2\pi j f k \Delta}$$

is the same as the definition of the discrete Fourier transform of the sequence $x_k(\sigma)$. Thus, one can take the FFT of the sequence $x_k(\sigma)$ and obtain $C(f, \sigma)$ for all frequencies that are integer multiples of $1/T$. In this case, and with the choice $\Delta_f = 1/(nT)$, as in Eq. (6), one actually needs the transform at a finer frequency resolution, finer by a factor of n . (For

example, if $n = 4$, the frequency resolution needs to be finer by a factor of 4.) This resolution can be obtained by taking the original sequence, of length N , and appending $(n - 1)N$ zeroes at the end, to obtain a sequence of length nN , and then computing the transform of the longer sequence. The overall complexity of this method is that of forming the FFT of a sequence of length nN , which is $cnN \log(nN)$.

For the case of multiple (say, M) candidate signal parameters σ , the process has to be repeated M times, resulting in complexity of $cnMN \log(nN)$.

In the special case where $x_k(\sigma) = y_k s_{k-\sigma}$, no further simplification appears possible and the complexity remains $cnMN \log(nN)$, which can also be prohibitively large. Furthermore, after performing this method, one does not have a computationally convenient way to adjust the sum to another nearby frequency.

NONCOHERENT PROCESSING

Another common way of avoiding the need for a large number of candidate frequencies is to process the data record non-coherently. A typical approach is the following. Break the data record x_1, \dots, x_N into a number N/N_0 (assumed integer) of blocks, consisting of N_0 data points each. For each data block, use one of the preceding methods to obtain an ambiguity function $|C_i(f, \sigma)|^2$ based on the data in the i th block. One then combines noncoherently the results from the various blocks, to obtain an overall ambiguity function given by

$$A(f, \sigma) = \sum_{i=1}^{N/N_0} |C_i(f, \sigma)|^2.$$

This method can be advantageous because coherence only needs to be maintained over intervals of length $N_0\Delta$ (the length of a block), as opposed to $N\Delta$. Hence, the number

of candidate frequencies, and the corresponding computational complexity is reduced by a factor of N/N_0 . In the absence of noise, this method delivers results comparable to the coherent method with much less computation.

The disadvantage of this method lies in the fact that it is much more sensitive to noise.

5 In many settings, coherent processing is optimal, as far as its statistical properties are concerned. Non-coherent processing suffers a so-called “squaring loss” that results in reduced sensitivity. This loss becomes more pronounced when the signal is weak and a long data record is needed.

The invention described in this application aims at reducing computational requirements for the case of long data records. Long data records are needed mainly when the signal is weak and the signal-to-noise ratio (SNR) is low. In this context, the noise sensitivity of non-coherent processing is very pronounced. For the non-coherent method to achieve a sensitivity comparable to that of the coherent one, the data set would have to be much larger, and there would be no net decrease in computational requirements.

15 SUMMARY OF THE INVENTION

Techniques are provided for synthesizing a long coherent I and Q correlation sum or integral at a particular frequency by synthetically combining a sequence of shorter correlation sums or integrals at the same or different frequency. Techniques are also provided for acquiring a carrier-modulated signal with an unknown shift of the carrier frequency, and possibly some additional unknown signal parameters. These techniques involve synthesizing coherent correlation sums or integrals at a fine frequency resolution, using coherent correlation sums or integrals that are calculated at a coarse frequency resolution. This approach allows

for coherent processing of the received signal over an arbitrarily long time interval, while avoiding the excessive computational requirements of traditional methods.

The method provides an approximation $D(f, \sigma)$ to the correlation sum

$$C(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j f k \Delta + j \theta_k},$$

5 for various candidate values of the pair (f, σ) , and uses these approximate values to acquire the signal, i.e., identify the true value of the carrier frequency and of signal parameter σ .

In the special case where σ is a delay parameter, and $s_k(\sigma)$ is of the form $s_{k-\sigma}$, further efficiency improvements are obtained by using Fast Fourier Transform techniques. A similar approximation is provided in the case where it is desired to approximate correlation integrals
10 instead of correlation sums.

The central idea is to synthesize long coherent correlation sums, at a fine set of frequencies, by combining short coherent correlation sums at a coarser set of frequencies. Specifically, if we split the index set $\{1, \dots, N\}$ into blocks $\{k_i, \dots, k_{i+1} - 1\}$ and if we calculate short correlation sums

$$15 \quad D_i(f', \sigma) = \sum_{k=k_i}^{k_{i+1}-1} y_k s_k(\sigma) e^{2\pi j f' k \Delta}$$

for some frequency f' , and for each block $\{k_i, \dots, k_{i+1} - 1\}$, then we can synthesize a very good approximation $D(f, \sigma)$ of the long correlation sum $C(f, \sigma)$ at a nearby frequency f , by letting

$$D(f, \sigma) = \sum_i e^{2\pi j (f - f') k_i \Delta + j \theta_{k_i}} D_i(f', \sigma).$$

20 The resulting approximation $D(f, \sigma)$ is of the form

$$D(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j \phi_f(k) \Delta},$$

where $\phi_f(k)$ is a time-dependent phase such that $|2\pi\phi_f(k)\Delta - 2\pi fk\Delta - \theta_k|$ is small for all k . This results in $D(f, \sigma)$ being a close approximation of $C(f, \sigma)$. The time-dependent phase $\phi_f(k)$ has a particular piecewise linear form, and is linear inside each block. In some embodiments, the method includes a procedure that selects certain parameters and a specific choice of $\phi_f(k)$ that optimizes the overall running time.

PERFORMANCE LOSSES

With proper choice of the algorithm parameters and of the time-dependent phase $\phi_f(k)$, this method does not suffer from coherence loss any more than the classical approaches do. A justification of this statement follows.

Recall that in classical methods for frequency search, one chooses a frequency grid spacing Δ_f so that $\Delta_f = 1/nT = 1/(nN\Delta)$, where n is a user-selected parameter. This ensures the following property. For every possible frequency f^* , there is a frequency f on the selected grid, for which $C(f, \sigma)$ is calculated, and for which $|f - f^*| \leq 1/(2nN\Delta)$, resulting in

$$|2\pi fk\Delta - 2\pi f^*k\Delta| \leq \frac{2\pi}{2n}. \quad \text{for all } k.$$

This method will guarantee the following analogous property. For every possible frequency f^* , there is a frequency f on the selected grid, for which $D(f, \sigma)$ is calculated, and for which

$$|2\pi\phi_f(k)\Delta - 2\pi f^*k\Delta - \theta_k| \leq \frac{2\pi}{n}, \quad \text{for all } k.$$

Thus, if one chooses n in the method to be twice as large as the value of n used in a classical method, the method provides an approximation of the true phase $2\pi f^*k\Delta + \theta_k$ that has the same guarantees as those provided by classical methods.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

5 FIG. 1A is a block diagram that illustrates a system overview for processing a signal; FIG. 1B is a flowchart that illustrates the operational overview of a technique for synthesizing coherent sums and for using them for the purpose of carrier estimation and signal acquisition.

10 FIG. 2A and FIG. 2B are flowcharts that provide an alternative illustration of a technique for estimating the carrier frequency of a received signal;

FIG. 2C is a flowchart that illustrates the details of the process for choosing a particular instantiation of the method and its associated parameters;

FIG. 3 illustrates a structure of a coarse-grained grid;

15 FIG. 4 is a flowchart depicting the details of the process for calculating the block-level correlations;

FIG. 5 is a flowchart depicting the details of the process for generating the reference signal;

FIG. 6 is a flowchart depicting the details of the process for combining the block-level correlations to obtain the correlation function $D(f, \sigma)$;

20 FIG. 7 illustrates the frequencies of the form $f_h + \hat{f}_q$ comprising the set \mathcal{F} ; and FIG. 8 is a block diagram illustrating a computer system on which embodiments of the invention may be implemented.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Techniques are provided for synthesizing coherent correlation sums or integrals at a fine frequency resolution, using coherent correlation sums or integrals that are calculated at a coarse frequency resolution. This approach allows for coherent processing of the received signal over an arbitrarily long time interval, while avoiding the excessive computational requirements of traditional methods. In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention.

FUNCTIONAL OVERVIEW

FIG. 1A is a block diagram that illustrates a system overview for processing a signal. The system comprises a signal source S, a receiver H, and a server D. By way of example, the signal source is a machine that produces an analog signal. The analog signal is known, except that it depends on an unknown signal parameter σ in a known manner. The analog signal is transmitted to receiver H. The signal that is received at H is herein referred to as a “received signal”. In some embodiments the unknown signal parameter σ is an unknown delay in time from the time the analog signal leaves signal source S and is received at receiver H. In one embodiment of the invention, H converts the analog signal into a sequence of numbers by filtering and then sampling the received signal. The sampled received signal is herein referred to as “sampled data” or simply “data.” In one embodiment of the invention, H transmits

the sampled data to server D for processing. Further, for the purpose of explanation, assume that the known signal's carrier frequency is modulated by an unknown frequency shift, which may, for example, be due to a clock drift at the receiver, and/or relative motion of the receiver with respect to the signal source. Thus, it is assumed that the sampled data received at 5 server D has an unknown carrier frequency due to the unknown frequency shift, and that the frequency shift belongs to a known frequency range $[F_1, F_2]$.

The present invention is equally applicable to the case where correlation integrals (instead of correlation sums) are to be computed, possibly in hardware, using an analog received signal, without first sampling the received signal. For ease of exposition, we describe an 10 embodiment for the case where calculations are based on sampled data. However, one skilled in the art would have no difficulty in making the necessary adjustments for the case of an analog received signal.

In FIG. 1B as explained herein, the method receives certain input parameters 100, which consist of the following. N is the length of the data record (number of samples) to be 15 processed. Δ is the real time elapsed between consecutive samples in the data record. F_1 and F_2 specify the frequency range to be searched; in particular, the set of frequencies f of interest is specified by $F_1 \leq f \leq F_2$. The parameter n is a positive integer that specifies a tolerance parameter for phase errors. Given n , the algorithm will ensure that phase errors due to approximations do not exceed $2\pi/n$ radians. In particular, the method will examine 20 frequencies that belong to a grid, where the distance between consecutive grid points is approximately $1/(nN\Delta)$ Hz. M is a positive integer that specifies the number of different candidate signal parameters σ that will be examined. DELAY is a binary parameter. If

DELAY is TRUE, the candidate parameters σ are integer and the reference signals $s_k(\sigma)$ are of the form $s_{k-\sigma}$, so that σ represents time delay. If DELAY is FALSE, then no assumption on the structure of the $s_k(\sigma)$ can be made. In either case, and without loss of generality, one can index the candidate signal parameters σ so that they range from 1 to M . The notation 5 $\mathcal{M} = \{1, \dots, M\}$ is used to refer to the set of candidate σ . Finally, c is a positive number that reflects the computational complexity of performing Fast Fourier Transforms (FFTs). In particular, the complexity of performing an FFT or an inverse FFT on a (possibly complex) sequence of length n will be estimated to be $cn \log n$.

FIG. 1B is a flowchart that illustrates the operational overview of a technique for synthesizing coherent sums. The process of block 200 is described in greater detail in FIG. 2C. At block 200 of FIG. 1B, a particular instantiation of the method is selected, so as to optimize its computational requirements. Block 200 receives input parameters 100. The instantiation of the method is specified by certain algorithm parameters 250. These algorithm parameters include the length N_0 of the blocks into which the data are broken, the number L of such 15 blocks, the number H of different coarse-grained frequencies that will be considered while processing each block, and the set of these frequencies f_1, \dots, f_H . The latter frequencies form a grid with constant spacing between consecutive grid points. The last algorithm parameters are two binary variables, BLOCK-FFT and COMB-FFT. If BLOCK-FFT is TRUE, then an FFT-based algorithm is to be used during the process at block 400 of processing each 20 block. The details of process at block 400 are described in greater detail in FIG. 4 herein. If COMB-FFT is TRUE, then an FFT-based algorithm is to be used by the process at block 800 of combining the results from the various blocks. The details of process at block 800 is

described in further detail in FIG. 6 herein.

The data y_1, \dots, y_N to be processed, shown in block 300, are fed to a process at block 400 that calculates for each block l , for $l = 1, \dots, L$, for each frequency f_h in the set f_1, \dots, f_H , and for each σ in the set $\mathcal{M} = \{1, \dots, M\}$, the block-level correlations at block 700, according

5 to the formula

$$u_{l,h}(\sigma) = \sum_{k=(l-1)N_0+1}^{lN_0} y_k s_k(\sigma) e^{2\pi j f_h(k-(l-1)N_0)\Delta}.$$

To carry out this task, the process at block 400 submits requests to a process at block 600 that generates the required values 500 of the reference signals $s_k(\sigma)$. The process at block 600 is described in greater detail in FIG. 5 herein.

10 The block-level correlations are complex-valued. Their real and imaginary parts are known as I and Q correlation sums. Accordingly, a block-level correlation will also be referred to as a “pair of I and Q correlation values” or as a “pair of I and Q correlation sums” or as a “pair of I and Q correlation integrals.”

15 The process at block 800 first constructs a finite set \mathcal{F} of equispaced fine-grained frequencies to be examined, of cardinality at most LH . These frequencies are constructed by refining the frequency grid f_1, \dots, f_H of the algorithm parameters 250 by a factor of L , that is, by reducing the spacing between consecutive grid points by a factor of L , while omitting some of the frequencies that fall outside the frequency range $[F_1, F_2]$. A more detailed description of the construction of \mathcal{F} is provided later.

20 The process at block 800 takes the block-level correlations $u_{l,h}(\sigma)$ as inputs and combines them, by forming weighted sums of the correlation sums $u_{l,h}(\sigma)$, to produce for each frequency f in the set \mathcal{F} , and for each candidate signal parameter σ in $\mathcal{M} = \{1, \dots, M\}$, an

approximation 850 $D(f, \sigma)$ to the correlation sum $C(f, \sigma)$, so that

$$D(f, \sigma) \approx C(f, \sigma) = \sum_{k=1}^N x_k s_k(\sigma) e^{2\pi j f k \Delta + j \theta_k}.$$

These approximate correlation sums are herein also referred to as weighted sums of correlation functions and are squared by the process at block 880 to obtain the approximate 5 ambiguity function 900 $|D(f, \sigma)|^2$, for every $f \in \mathcal{F}$ and every $\sigma \in \mathcal{M}$. Once this approximate ambiguity function becomes available for every $f \in \mathcal{F}$ and every $\sigma \in \mathcal{M}$, a final process at block 950 is used to select a pair (f, σ) at which the approximate ambiguity is maximized.

In one embodiment, the maximizing pair (f, σ) is selected when a single best estimate of an unknown frequency and signal parameter is to be reported. However, different 10 embodiments are possible. In some embodiments, the process may quit after process at block 800 and feed the approximate correlation sums $D(f, \sigma)$ to some other method for further processing. In other embodiments, the process may quit after process at block 880 and feed the approximate ambiguity function $|D(f, \sigma)|^2$ to some other method for further processing. Such other methods may employ additional side information, or may perform a 15 finer-grained search by interpolating the function $D(f, \sigma)$ as described in U.S. Patent Application No. _____, entitled “Extracting Fine-Tuned Estimates from Correlation Functions Evaluated at a Limited Number of Values” by Anant Sahai, John Tsitsiklis, Stefano Casadei, Andrew Chou, Benjamin Van Roy and Jesse Robert Stone, filed on the same day herewith, Attorney Docket No. 60021-0013. Finally, in some embodiments, the process 20 at block 950 may select all pairs (f, σ) that exceed a certain threshold. Or the process at block 950 may select a fixed number of pairs (f, σ) with the highest ambiguity values, for

further processing.

FIG. 2A and FIG. 2B are block diagrams that provide an alternative illustration of a technique for estimating the carrier frequency of a received signal. At block 202 of FIG. 2A, a frequency range of interest is determined. A frequency range of interest is the range of frequencies within which it is assumed that the true carrier frequency lies. In one embodiment of the invention, the frequency range of interest represents a pre-selected frequency interval around the carrier frequency of the known analog signal produced by signal source S.

At block 204, the frequency range is divided into multiple sets of frequency intervals, generally known as frequency bins. In one embodiment, two sets of frequency bins are used. The first set of frequency bins comprises a set of coarse grain frequency bins while the second set of frequency bins comprises a set of fine grain frequency bins. According to certain embodiments of the invention, the number of coarse grain frequency bins is preset to a constant which is independent of the length of the sampled data. In other embodiments, the number of coarse grain frequency bins is chosen to be proportional to the square root 15 of the length of the sampled data. In addition, in certain embodiments, the selection of the number of coarse grain frequency bins may be based upon a user-selected signal-to-noise ratio tolerance.

At block 206, the sampled data is divided into a set of data blocks. In one embodiment, the set of data blocks corresponds to the set of coarse grain frequency bins, in the sense 20 that the length of the data blocks is chosen by taking into account the size of the coarse grain frequency bins. At block 208, a range of hypothesized signal parameters such as delay values are determined by estimating the range of possible delay in time that is measured

from the time the signal is transmitted from the signal source S until the time that the signal is received at receiver H. At block 210, for each coarse grain frequency bin, a trial value of frequency is selected. In some embodiments the trial value is the midpoint of a frequency bin. The trial value of frequency is herein referred to as a “trial f ”. At block 212, control is

5 passed to block 220 of FIG. 2B.

At block 220 of FIG. 2B, for each data block, using the selected trial f , the In Phase and Quadrature correlation sums (“ I and Q correlation sums”) are calculated over the range of hypothesized delay values. At block 222, for each fine grain frequency bin and for each hypothesized delay value, the I and Q correlation sums are suitably weighted and then summed over all the data blocks, without re-calculating the I and Q correlation sums.

10 At block 224, if \bar{I} and \bar{Q} represent the summation of the I and Q sums respectively, then for each fine gain frequency bin and for each hypothesized delay value, the magnitude of the vector (\bar{I}, \bar{Q}) is calculated. At block 226, the fine grain frequency bin that contains the highest magnitude value is determined. The frequency value corresponding to the frequency 15 bin that has the highest magnitude value is estimated to be the carrier frequency of the received signal, which includes the Doppler shift, if any.

SELECTION OF AN INSTANTIATION OF THE METHOD

FIG. 2C is a flowchart that illustrates the details of the process at 200 of FIG. 1B for choosing a particular instantiation of the method and its associated parameters. The process 20 begins at block 228 of FIG. 2C by generating a set of candidate block lengths N_0 . In one embodiment, the set of candidate block lengths consists of all positive integers that are powers of 2 and that are smaller than N . In alternative embodiments the set of candidate

*N*₀ could be any set consisting of integers in the range {2, . . . , *N* − 1}.

At block 229 for each candidate block length *N*₀, calculate

$$H = \lceil n(F_2 - F_1)N_0\Delta \rceil,$$

which will be the number of frequencies that will be considered for individual blocks and

5 $L = \left\lceil \frac{N}{N_0} \right\rceil,$

which will be the corresponding number of blocks.

The next steps will rely on the following estimates of the computational complexity for different instantiations of the processes at block 400 and block 800 of FIG. 1B. These estimates are justified later, when the processes at block 400 and at block 800 are described 10 in more detail herein in FIG. 4 and FIG. 6 respectively.

If block-level correlations are to be calculated by process at block 400 in a default manner, the computational effort of process at block 400 of FIG. 1B is estimated as

6 $MHN.$

If block-level correlations are to be calculated by process at block 400 of FIG. 1B using

15 an FFT-based method, the computational effort of process at block 400 is estimated as

$$3c\frac{N}{N_0}H(N_0 + M)\log(N_0 + M)$$

This option is available only if $s_k(\sigma)$ is of the form $s_{k-\sigma}$, that is, if the input DELAY is TRUE.

If the combining of the block-level correlations is to be carried out by process at block 20 800 of FIG. 1B in a default manner, the computational effort of process at block 800 is

estimated as

$$\frac{MHN^2}{N_0^2}$$

If the combining of the block-level correlations is to be carried out by process at block 800 using an FFT-based method, the computational effort of process at block 800 is estimated 5 as

$$cnHM \frac{N}{N_0} \log \left(\frac{nN}{N_0} \right).$$

At block 230 of FIG. 2C, it is determined whether DELAY is FALSE or TRUE.

If DELAY=FALSE, the process at block 240 of FIG. 2C is used, which does the following.

For each candidate value of N_0 and associated value of H , it evaluates

$$MHN + \min \left\{ \frac{MHN^2}{N_0^2}, cnHM \frac{N}{N_0} \log \left(\frac{nN}{N_0} \right) \right\},$$

and selects that candidate value of N_0 for which the above expression is smallest. The above expression consists of the estimated complexity of calculating the block-level correlations in the default manner plus the estimated complexity of the better of the two alternative available methods of combining the block-level correlations. Furthermore, if the selected 15 value of N_0 , and the associated value of H , satisfy $\frac{MHN^2}{N_0^2} < cnHM \frac{N}{N_0} \log \left(\frac{nN}{N_0} \right)$, then the binary variable COMB-FFT is set to FALSE. Otherwise, COMB-FFT is set to TRUE.

If DELAY=TRUE, the process at block 250 is used, which does the following. For each candidate value of N_0 and associated value of H , it evaluates

$$\min \left\{ MHN, 3c \frac{N}{N_0} H(N_0 + M) \log(N_0 + M) \right\} + \min \left\{ \frac{MHN^2}{N_0^2}, cnHM \frac{N}{N_0} \log \left(\frac{nN}{N_0} \right) \right\}$$

20 and selects that candidate value of N_0 for which the above expression is smallest. The above expression consists of the estimated complexity of calculating the block-level correlations

using the better of the two alternative available methods plus the estimated complexity of combining the block-level correlations using the better of the two alternative available methods. Furthermore, if the selected value of N_0 , and the associated value of H , satisfy $MHN < 3c\frac{N}{N_0}H(N_0 + M)\log(N_0 + M)$, then the binary variable BLOCK-FFT is set to

5 FALSE. Otherwise, BLOCK-FFT is set to TRUE. Finally, if the selected value of N_0 , and the associated value of H , satisfy $\frac{MHN^2}{N_0^2} < cnHM\frac{N}{N_0}\log\left(\frac{nN}{N_0}\right)$ then the binary variable COMB-FFT is set to FALSE. Otherwise, COMB-FFT is set to TRUE.

At the conclusion of process at block 240 or at block 250, a particular value for N_0 and an associated value of H and L have been selected. Then, the process at block 260 chooses 10 the coarse-grained grid of frequencies f_1, \dots, f_H by letting

$$f_h = F_1 + \frac{1}{2N_0\Delta n} + (h-1)\frac{1}{N_0\Delta n} \quad h = 1, \dots, H.$$

Because of the way that H and f_H are chosen, the highest frequency on the grid is at least $F_2 - 1/(2N_0\Delta n)$. This is because

$$H \geq n(F_2 - F_1)N_0\Delta$$

15 and

$$f_H = F_1 + \frac{1}{2N_0\Delta n} + (H-1)\frac{1}{N_0\Delta n} \geq F_1 + \frac{1}{2N_0\Delta n} + F_2 - F_1 - \frac{1}{N_0\Delta n}.$$

Since the distance between consecutive grid points is $1/(N_0\Delta n)$, it follows that for every $f \in [F_1, F_2]$ there is some f_h which is within $1/(2N_0\Delta n)$ from f .

The structure of this grid is illustrated in FIG. 3. In that figure, and for the purpose of 20 illustration, it is assumed that $F_1 = 5000$, $F_2 = 9200$, and $N_0\Delta n = 1/1000$. This leads to $H = 5$ and $f_1 = 5500$, $f_2 = 6500$, $f_3 = 7500$, $f_4 = 8500$, $f_5 = 9500$.

CALCULATION OF BLOCK-LEVEL CORRELATIONS

FIG. 4 is a flowchart depicting the details of process at block 400 of FIG. 1B for calculating the block-level correlations. For each block $l \in \{1, \dots, L\}$, for each coarse-grained frequency $f_h, h \in \{1, \dots, H\}$, and for each $\sigma \in \mathcal{M} = \{1, \dots, M\}$, this process calculates the block-level correlation sums

$$u_{l,h}(\sigma) = \sum_{k=(l-1)N_0+1}^{lN_0} y_k s_k(\sigma) e^{2\pi j f_h(k-(l-1)N_0)\Delta}.$$

The process first examines at block 410 of FIG. 4 the algorithm parameter **DELAY**. If the algorithm parameter **DELAY** is FALSE, the process receives at block 420 $s_1(\sigma), \dots, s_N(\sigma)$ from the reference signal generator, and calculates at block 430 the desired $u_{l,h}(\sigma)$ by straight-forward multiplications and summations. The total computational effort of this step is estimated as

$$MHN.$$

If the algorithm parameter **DELAY** is TRUE, the process receives at block 440 s_{-M+1}, \dots, s_N from the reference signal generator and calculates for each block $l \in \{1, \dots, L\}$, for each frequency $f_h, h \in \{1, \dots, H\}$, and for each $\sigma \in \mathcal{M} = \{1, \dots, M\}$,

$$u_{l,h}(\sigma) = \sum_{k=(l-1)N_0+1}^{lN_0} y_k s_{k-\sigma} e^{2\pi j f_h(k-(l-1)N_0)\Delta},$$

after choosing at block 450 between one of two alternative methods. Thus, at block 450, it is determined whether **BLOCK-FFT** is TRUE or FALSE.

If **BLOCK-FFT** is FALSE, it proceeds using at block 460 straightforward multiplications and summations, and the total computational effort of this step is estimated as

$$MHN.$$

Alternatively, if BLOCK-FFT is TRUE, it uses a process at block 470 that relies on the fact that for any fixed l and h , the quantities $u_{l,h}(\sigma)$ correspond to a convolution of the sequence $s_{(l-1)N_0-M+1}, \dots, s_{lN_0}$, of length $N_0 + M$, with the sequence $y_k e^{2\pi j f_h(k-(l-1)N_0)\Delta}$, for $k = (l-1)N_0 + 1, lN_0$, of length N_0 . This convolution is calculated by padding both sequences 5 with zeroes at the end, so that their length becomes the same and equal to $N_0 + M$, taking the FFT of the two sequences, multiplying the FFTs, and then taking the inverse FFT.

The complexity of computing a single convolution is estimated as $3c(N_0 + M) \log(N_0 + M)$, where the factor of 3 is due to the fact that there is a total of 3 FFTs or inverse FFTs. There are LH convolutions to be computed (one for each (l, h) pair) and the total computational effort of this step is estimated as 10

$$3c \frac{N}{N_0} H(N_0 + M) \log(N_0 + M).$$

Alternative embodiments of the calculation at block 400 of FIG. 1B for block-level correlations are possible, especially when the reference signal has a special structure. For example, when processing GPS signals, the reference signal is a periodic PRN (pseudorandom noise) code, modulated by a slow binary signal. This structure is exploited in U.S. Patent Application No. _____, entitled “SIGNAL ACQUISITION USING DATA BIT INFORMATION” by Anant Sahai, Wallace Mann, Andrew Chou and Benjamin Van Roy, filed on the same day herewith, Attorney Docket No. 60021-0011. 15

Alternative embodiments are also possible in the case where the received signal is not sampled, but is processed in analog form. In that case, the block-level correlations $u_{l,h}(\sigma)$ 20 can be calculated using one or more analog correlators, implemented in hardware.

GENERATING THE REFERENCE SIGNAL

FIG. 5 is a flowchart depicting the details of the process at block 600 of FIG. 1B for generating the reference signal. At block 510 of FIG. 5, it is determined whether the parameter DELAY is TRUE or FALSE. If algorithm parameter DELAY is FALSE, then at block 520 of FIG. 5 the reference signal generator stores in random access memory a sequence of values $s_1(\sigma), \dots, s_N(\sigma)$, for every $\sigma \in \mathcal{M}$ and supplies these values as needed by process at block 400 of FIG. 1B. If algorithm parameter DELAY is TRUE, then at block 530 the reference signal generator stores in its memory a sequence of values s_{-M+1}, \dots, s_N and supplies these values as needed by process at block 400. In particular, for every k and σ , $s_k(\sigma) = s_{k-\sigma}$ is set.

In alternative embodiments, the reference signal is not read from memory but is generated on the fly by a computational algorithm.

In alternative embodiments, the reference signal is calculated by combining information that is stored in local memory with other information that is externally provided. For instance, if the reference signal is a GPS signal, it can be formed by modulating a locally stored PRN code by a slowly varying binary signal consisting of “navigation bits.” These navigation bits are not a priori known, but can be obtained in real time from a commercial provider or from a differential GPS station via a communication network, such as the internet.

COMBINING THE BLOCK-CORRELATIONS

FIG. 6 is a flowchart depicting the details of the process at 800 of FIG. 1B for combining the block-level correlations to obtain the correlation function $D(f, \sigma)$, for every $\sigma \in \mathcal{M}$ and for every f in a certain set \mathcal{F} of fine-grained frequencies that is constructed in the beginning of process at block 800.

At block 810 of FIG. 6 a set of frequencies $\hat{f}_1, \dots, \hat{f}_L$, is constructed using the formula

$$\hat{f}_q = -\frac{1}{2nN_0\Delta} + \frac{1}{2LnN_0\Delta} + \frac{q-1}{L} \frac{1}{nN_0\Delta}, \quad q = 1, \dots, L.$$

Note that these frequencies are spaced $1/(nLN_0\Delta)$ apart, which is a factor of L smaller than the spacing $1/(nN_0\Delta)$ of the coarse-grained frequencies f_1, \dots, f_H . Subsequently, at block 5 820 the process forms the set of all frequencies of the form

$$f_h + \hat{f}_q,$$

where h ranges over the set $\{1, \dots, H\}$ and q ranges over the set $\{1, \dots, L\}$. Then, at block 830, the above set is pruned by eliminating those frequencies $f_h + \hat{f}_q$ that are larger than $F_2 + (1/nN_0\Delta L)$. Let \mathcal{F} be the resulting set, which is the set of fine-grained frequencies to be considered. Its cardinality is denoted by B and satisfies $B \leq LH$.

The smallest frequency in that set is

$$f_1 + \hat{f}_1 = F_1 + \frac{1}{2N_0\Delta n} - \frac{1}{2N_0\Delta n} + \frac{1}{2LN_0\Delta n} = F_1 + \frac{1}{2LN_0\Delta n}.$$

The largest frequency of the form $f_h + \hat{f}_q$ is

$$\begin{aligned} f_H + \hat{f}_L &= F_1 + \frac{1}{2N_0\Delta n} + (H-1) \frac{1}{N_0\Delta n} - \frac{1}{2nN_0\Delta} + \frac{1}{2LN_0\Delta n} + \frac{L-1}{L} \frac{1}{nN_0\Delta} \\ 15 &\geq F_2 - \frac{1}{2nN_0\Delta L}. \end{aligned}$$

Because of the choice of the spacing, which is $1/(nN_0\Delta L)$, it follows that for every $f^* \in [F_1, F_2]$, there exists some $f_h + \hat{f}_q \in \mathcal{F}$ which is within $1/(2nN_0\Delta L)$ of f^* .

The frequencies of the form $f_h + \hat{f}_q$ comprising the set \mathcal{F} are illustrated in FIG. 7. In FIG. 7, and for the purpose of illustration, it is assumed that $F_1 = 5000$, $F_2 = 9200$,

$N_0\Delta n = 1/1000$, and $L = 4$. In particular, $H = 5$ and the spacing between consecutive fine-grained frequencies is $1/(nLN_0\Delta) = 250$. The fine-grained frequencies are indicated by the heavy dots on the frequency axis. For reference, the coarse-grained frequencies f_h are also indicated. The lowest fine-grained frequency is $F_1 + 1/(nLN_0\Delta) = 5125$ and the highest 5 is 9375. The cardinality B of \mathcal{F} is 18, which is somewhat smaller than $LH = 20$.

The rest of this process at block 800 calculates

$$D(f_h + \hat{f}_q, \sigma) = \sum_{l=1}^L u_{l,h}(\sigma) e^{2\pi j(l-1)(f_h + \hat{f}_q)N_0\Delta + j\theta_{(l-1)N_0}},$$

for every $f_h + \hat{f}_q \in \mathcal{F}$. The calculation uses as input 840 the numbers $u_{l,h}(\sigma)$ (one such number for each $l \in \{1, \dots, L\}$, each $h \in \{1, \dots, H\}$, and each $\sigma \in \mathcal{M}$) which are the output 10 of the process at block 400 of FIG. 1B. At Block 845, it is determined whether the binary algorithm parameter COMB-FFT has been set by process at block 200 of FIG. 1B to TRUE or FALSE.

If COMB-FFT is FALSE, the process at block 850 that is used to calculate $D(f_h + \hat{f}_q, \sigma)$ just implements the above displayed formula. This is done for M values of σ , and B values of 15 $f_h + \hat{f}_q$. For each $(f_h + \hat{f}_q, \sigma)$ pair, there are $L \approx N/N_0$ complex additions and multiplications to be performed. Thus, the complexity of the process at block 850 can be estimated as

$$MB \frac{N}{N_0}.$$

Typically, one has $B/N \approx H/N_0$, or $B \approx NH/N_0$, and the above complexity estimate becomes

$$\frac{MHN^2}{N_0^2}$$

20

If COMB-FFT is TRUE, an alternative process at block 860 is used. It is observed that

$D(f_h + \hat{f}_q, \sigma)$ can be rewritten as

$$D(f_h + \hat{f}_q, \sigma) = \sum_{l=1}^L u_{l,h}(\sigma) e^{2\pi j(l-1)f_h N_0 \Delta + j\theta_{(l-1)N_0}} e^{2\pi j(l-1)\hat{f}_q N_0 \Delta}$$

Viewed as a function of $\hat{f}_q N_0 \Delta$, this is the discrete Fourier transform of the sequence

5 $u_{l,h}(\sigma) e^{2\pi j(l-1)f_h N_0 \Delta + j\theta_{(l-1)N_0}}$, $l = 1, \dots, L$. Thus, one can take the FFT of this sequence and obtain $D(f_h + \hat{f}_q, \sigma)$ for all \hat{f}_q such that $\hat{f}_q N_0 \Delta$ is an integer multiple of the fundamental frequency which is $1/L$. That is the same as computing $D(f_h + \hat{f}_q, \sigma)$ for all \hat{f}_q that are integer multiples of $1/(LN_0 \Delta)$. However, the frequencies \hat{f}_q of interest are spaced $1/(nLN_0 \Delta)$ apart, and therefore, the frequency resolution has to be refined by a factor of n .

10 The above is accomplished by the process at block 860 which does the following. Take the original sequence $u_{l,h}(\sigma) e^{2\pi j(l-1)f_h N_0 \Delta + j\theta_{(l-1)N_0}}$, $l = 1, \dots, L$, of length L , append $(n-1)L$ zeroes at the end, to obtain a sequence of length nL , and then compute the FFT of the longer sequence. For each $\sigma \in \mathcal{M}$ and for each f_h , the transform at the required resolution is computed with $cnL \log(nL)L$ operations. Since $L \approx N/N_0$, the complexity of this process

15 is estimated as

$$cnHM \frac{N}{N_0} \log \left(\frac{nN}{N_0} \right).$$

APPROXIMATION GUARANTEES

In this subsection, it is explained why the method described in the above has the desired approximation guarantees. For simplicity, let us consider the case where the additional phase
20 term θ_k is absent. The quantity approximated by the method is given by

$$C(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j f k \Delta}.$$

The approximation constructed is of the form

$$D(f, \sigma) = \sum_{k=1}^N y_k s_k(\sigma) e^{2\pi j \phi_f(k)\Delta}.$$

For k in the l th block, one has

$$\phi_f(k) = f_h(k - (l-1)N_0) + (l-1)(f_h + \hat{f}_q)N_0 = (f_h + \hat{f}_q)k - \hat{f}_q(k - (l-1))N_0.$$

5 Indeed, the first term $f_h(k - (l-1)N_0)$ is introduced when computing the block-level correlations

$$u_{l,h}(\sigma) = \sum_{k=(l-1)N_0+1}^{lN_0} y_k s_k(\sigma) e^{2\pi j f_h(k-(l-1)N_0)\Delta}.$$

The second term $(l-1)(f_h + \hat{f}_q)N_0$ is introduced by the combining formula

$$D(f_h + \hat{f}_q, \sigma) = \sum_{l=1}^L u_{l,h}(\sigma) e^{2\pi j(l-1)(f_h + \hat{f}_q)N_0\Delta + j\theta_{(l-1)N_0}},$$

10 For every possible frequency f^* in the range $[F_1, F_2]$, and because the frequencies $f_h + \hat{f}_q$ are spaced $1/(nN_0L\Delta)$ apart, there is some $f = f_h + \hat{f}_q$ such that

$$|f - f^*| \leq \frac{1}{2nN_0L\Delta}.$$

Further, it follows that

$$\begin{aligned} |f^*k - \phi_f(k)| &\leq |f^* - f_h - \hat{f}_q|k + |\hat{f}_q|(k - (l-1)N_0) \\ 15 &\leq \frac{1}{2nN_0L\Delta}N_0L + \frac{1}{2nN_0\Delta}N_0 \\ &= \frac{1}{n\Delta}. \end{aligned}$$

It follows that for every possible frequency f^* , there is a fine-grained frequency $f = f_h + \hat{f}_q$, for which $D(f, \sigma)$ is calculated, and for which

$$|2\pi\phi_f(k)\Delta - 2\pi f^*k\Delta| \leq \frac{2\pi}{n}, \quad \text{for all } k.$$

When the time-varying phase θ_k is present, there is an additional phase error equal to the maximum change of θ_k in the course of a block. When θ_k changes slowly, and provided that the block length is not too large, the error due to this additional phase term is negligible.

ALTERNATIVE EMBODIMENTS

5 The embodiments as described herein are easily modified to deal with certain changes in the objective of the computation or to accommodate certain preferences and other considerations. A number of alternative embodiments of the various steps are discussed in the sequel.

SELECTION OF COARSE AND FINE-GRAINED FREQUENCY GRIDS

10 There are many ways of selecting the grid of coarse frequencies f_h and the grid \mathcal{F} of fine-grained frequencies. In some embodiments, the density of the grid \mathcal{F} need not be L times more than the density of the initial grid f_1, \dots, f_H . In some embodiments, one or both of these grids can be chosen to have non-uniform spacing between consecutive grid points. In some embodiments, the frequency grid points are selected ahead of time by the
15 user, whereas in other embodiments the frequency grids are selected by the algorithm taking into account various input parameters. Finally, in some embodiments, one may choose the offsets between the coarse and the fine grids so that the coarse grid is a subset of the fine grid.

CHOOSING THE FREQUENCY GRID SPACINGS

20 In one embodiment, the spacing of the coarse and of the fine-grained frequency grids is indirectly specified by the input parameter n that reflects a tolerance for phase approximation errors. In alternative embodiments, the various grid spacings are directly specified by the

user. In other embodiments, one uses a formula that quantifies performance losses as a function of the spacing of the various grids and selects the grid spacing by evaluating the tradeoff between performance losses and computational complexity.

PARTITIONING THE FREQUENCY RANGE $[F_1, F_2]$.

5 In some embodiments, one can start by partitioning the frequency range $[F_1, F_2]$ into disjoint or overlapping frequency subintervals, and apply the method separately for each one of these subintervals. In some instances, the subintervals can be chosen small enough so that $H = 1$ and a single coarse-grained frequency will suffice for each subinterval. That single frequency can be chosen to be the center of the subinterval, or can be chosen differently.

10 SELECTION OF ALGORITHM PARAMETERS

The selection of a particular method during process at block 200 of FIG. 1B, can be made using alternative complexity criteria. For example, if certain FFTs (of the reference signal or of the data sequence) are available at no extra computation cost, (e.g., if they are calculated by some other part of a broader algorithm), then the corresponding computational

15 cost should not be included in the cost of process at block 400. Furthermore, if different embodiments of the processes at block 400 and at block 800 of FIG. 1B are employed, then their computational costs have to be estimated and used in selecting the most efficient method.

The particular complexity criteria used in the embodiment of process at block 200 of FIG.

20 1B can be replaced by other estimates of the run times of the various blocks, depending on the desired degree of precision. For example, the cardinality of the set \mathcal{F} was estimated as

LH. An alternative embodiment estimates the cardinality of \mathcal{F} as

$$\lceil n(F_2 - F_1)N\Delta \rceil,$$

which can be more accurate when $F_2 - F_1$ is small.

One embodiment chooses between FFT-based and default methods, and sets the binary

5 variables BLOCK-FFT and COMB-FFT based on computational complexity estimates. In
alternative embodiments, these choices are preset by the user.

In alternative embodiments, the method is applied in contexts where the binary input
variable DELAY is always TRUE, or always FALSE. In such cases, this variable need not
be treated as an input of the algorithm, and the method can be accordingly simplified by
10 omitting those alternatives that do not apply.

One embodiment provides a specific method for choosing the block size N_0 . In alternative
embodiments the block size can be preset by the user. Or the block size can be selected
according to another set of criteria. In some cases, there may be a restriction on the minimum
allowed block size. In some cases, there may be a restriction on the maximum allowed block
15 size. In some cases, the block length may be restricted to be approximately equal to the
square root of N .

MISSING DATA

Certain embodiments are easily modified to deal with the case in which some of the
data samples y_k are missing. In one embodiment, the calculations are performed only over
20 those blocks where there are no missing data. In an alternative embodiment, missing data
are set to zero, and the calculations are performed only over those blocks in which there

are nonzero data. The complexity calculations in process at block 200 of FIG. 1B can be suitably modified to take this variation into account.

A TOP-DOWN APPROACH

Certain embodiments use a bottom-up approach in which all block-level correlations are 5 computed first by the process at block 400 of FIG. 1B, and are then used to calculate the approximate correlation sums $D(f, \sigma)$ for all candidate f and σ . In an alternative embodiment, one may start by considering a limited set of candidate f and σ , and calculate $D(f, \sigma)$ only for the chosen candidates. This limited set may consist of just one or multiple candidates. The calculation of $D(f, \sigma)$ for some candidate (f, σ) requires certain quantities 10 $u_{l,h}(\sigma)$ to be made available. These quantities can be calculated as needed. Furthermore, once a quantity $u_{l,h}(\sigma)$ is calculated, it is stored in memory, so that if it is ever needed in the future it need not be recomputed. This variation is useful in a situation where certain pairs (f, σ) are examined first. If the ambiguity value associated with some pair (f, σ) turns out to be above some threshold, the process can stop. Otherwise, it can continue to examine a 15 larger set of pairs.

A SEQUENTIAL APPROACH

Certain embodiments commit to processing the full set of data. In an alternative embodiment, a smaller set of data is examined first. This can be accomplished, for example, by keeping the same block length N_0 but using a smaller number L' of blocks, so that only 20 $L'N_0 < N$ of the data are utilized. If the ambiguity value associated with some pair (f, σ) is found to exceed a certain threshold, the algorithm is deemed to have successfully acquired the signal and the computation can stop. Otherwise, a larger number of blocks is fed to the

algorithm, and the algorithm repeats on the expanded set of blocks and data. Any intermediate results that have been calculated when using a smaller number of blocks are saved in memory so that they can be reused later. When dealing with a smaller set of data, the grid-spacing of the frequencies to be searched can be taken to be coarser, with an attendant reduction of the computational effort. If the number of blocks is subsequently increased, the frequency grid is refined. With such an approach, the computational complexity of the overall method will be different, and suitable changes have to be made in the process used to select the various algorithm parameters.

A HIERARCHICAL VERSION

In certain embodiments, there are only two frequency grids and the data are broken into a single set of blocks. In alternative embodiments, the same method is used in a hierarchical manner. In one embodiment, the blocks are broken down into subblocks and the calculation of the various block-level correlations is approximated by calculating correlation sums for each subblock and then merging them together to synthesize block-level correlations. This decomposition can be continued recursively by introducing an arbitrary number of levels.

In the more general embodiment, a hierarchy of levels is used. With each level r there is an associated decomposition of the data into blocks, and an associated grid \mathcal{F}_r of frequencies. For each level r , the grid of frequencies \mathcal{F}_r need not be regularly spaced. The only constraint is on the block sizes at the different levels. A block at a higher level r must be the union of smaller blocks at lower levels $r' < r$. One then proceeds as follows. Starting from the lowest level, call it level 1, calculate correlation sums $D_i(f, \sigma)$ for the various blocks i and for the various frequencies f in the set of frequencies \mathcal{F}_1 associated with that level. For a higher

level $r > 1$, and for all the blocks and all the frequencies in the set \mathcal{F}_r associated with that level, calculate approximate correlation sums, as a linear combination of the correlation sums calculated from a lower level. In particular, the calculation of a correlation sum $D_i(f, \sigma)$ at a certain frequency $f \in \mathcal{F}_r$ and for a certain r -level block i , makes use of the correlation sums $D_j(f', \sigma)$ for those $(r-1)$ -level blocks j that comprise block i , and for those frequencies $f' \in \mathcal{F}_{r-1}$ that are closest to f .

EXPLORING A SMALL FREQUENCY RANGE

In some situations, one may want to evaluate approximate correlation sums $D(f, \sigma)$ for a small range of frequencies, or even at a single frequency. This is accomplished by forming a suitable linear combination of block-level correlations that are already available at frequencies in the vicinity of the frequency f of interest. An embodiment of this kind is of interest in various contexts. One particular context is when the signal has already been acquired, a moderately accurate estimate \hat{f} of the true carrier frequency is available, and a search for a more accurate estimate by exploring the vicinity of \hat{f} is desired.

15

HARDWARE OVERVIEW

An embodiment of the invention may be implemented using a computer system that includes a processor for processing information. The Computer system also includes a main memory, such as a random access memory (RAM) or other dynamic storage device, coupled to a bus for storing information and instructions to be executed by the processor. The main memory also may be used for storing temporary variables or other intermediate information during execution of instructions to be executed by the processor. The computer system further includes a read only memory (ROM) or other static storage device coupled to the

bus for storing static information and instructions for the processor. A storage device, such as a magnetic disk or optical disk, is provided and coupled to the bus for storing information and instructions.

The invention is related to the use of the computer system for implementing the techniques described herein. According to one embodiment of the invention, those techniques are implemented by the computer system in response to the processor executing one or more sequences of one or more instructions contained in main memory. Such instructions may be read into the main memory from another computer-readable medium, such as the storage device. Execution of the sequences of instructions contained in the main memory causes the processor to perform the process steps described herein. One or more processors in a multi-processing arrangement may also be employed to execute the sequences of instructions contained in the main memory. In alternative embodiments, hard-wired circuitry may be used in place of or in combination with software instructions to implement the invention. Thus, embodiments of the invention are not limited to any specific combination of hardware circuitry and software.

The term "computer-readable medium" as used herein refers to any medium that participates in providing instructions to the processor for execution. Such a medium may take many forms, including but not limited to, non-volatile media, volatile media, and transmission media. Non-volatile media includes, for example, optical or magnetic disks, such as the storage device. Volatile media includes dynamic memory, such as the main memory. Transmission media includes coaxial cables, copper wire and fiber optics, including the wires that comprise the bus. Transmission media can also take the form of acoustic or light waves,

such as those generated during radio wave and infrared data communications.

Common forms of computer-readable media include, for example, a floppy disk, a flexible disk, hard disk, magnetic tape, or any other magnetic medium, a CD-ROM, any other optical medium, punch cards, paper tape, any other physical medium with patterns of holes, a RAM, 5 a PROM, and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave as described hereinafter, or any other medium from which a computer can read.

Various forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to the processor for execution. For example, the

instructions may initially be carried on a magnetic disk of a remote computer. The remote 10 computer can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to computer system can receive the data on the telephone line and use an infrared transmitter to convert the data to an infrared signal. An infrared detector coupled to the bus can receive the data carried in the infrared signal and place the data on the bus. The bus carries the data to the main memory, from which 15 the processor retrieves and executes the instructions. The instructions received by the main memory may optionally be stored on the storage device either before or after execution by the processor.

The computer system also includes a communication interface coupled to the bus. The communication interface provides a two-way data communication coupling to a network link

20 that is connected to a local network. For example, the communication interface may be an integrated services digital network (ISDN) card or a modem to provide a data communication connection to a corresponding type of telephone line. As another example, the commun-

cation interface may be a local area network (LAN) card to provide a data communication connection to a compatible LAN. Wireless links may also be implemented. In any such implementation, the communication interface sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information.

5 The network link typically provides data communication through one or more networks to other data devices. For example, the network link may provide a connection through the local network to a host computer or to data equipment operated by an Internet Service Provider (ISP). The ISP in turn provides data communication services through the worldwide packet data communication network now commonly referred to as the "Internet". The local network and the Internet both use electrical, electromagnetic or optical signals that carry digital data streams. The signals through the various networks and the signals on the network link and through the communication interface, which carry the digital data to and from the computer system, are exemplary forms of carrier waves transporting the information.

The computer system can send messages and receive data, including program code,

15 through the network(s), the network link and the communication interface. In the Internet example, a server might transmit a requested code for an application program through the Internet, the ISP, the local network and the communication interface. In accordance with the invention, one such downloaded application implements the techniques described herein.

20 The received code may be executed by the processor as it is received, and/or stored in the storage device, or other non-volatile storage for later execution. In this manner, the computer system may obtain application code in the form of a carrier wave.

In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative rather than 5 a restrictive sense.

THE JOURNAL OF CLIMATE